



REST API

Transactional

Operations

10/28/2013

Contents

Introduction	3
Audience	3
Prerequisites.....	3
Acknowledgments.....	3
Scope.....	3
SIS Operations.....	4
Assumptions	4
Note about read-only entities	4
Operations	5

Introduction

This document defines the transactional operations in the Ed-Fi REST API that are required to satisfy the operations that occur through a Student Information System (SIS). The purpose of this document is to enable SIS vendors to easily integrate with the Ed-Fi REST API to read and write data.

Audience

This document is designed for application developers that are integrating with the Ed-Fi REST API. Since the Ed-Fi REST API is middleware, no reference is made to a user interface since that would be supplied by the SIS which would communicate to the Ed-Fi REST API through automated calls.

Prerequisites

To gain the most benefit from this information, you should already be familiar with Representational State Transfer (REST) services (http://en.wikipedia.org/wiki/Representational_state_transfer).

Acknowledgments

This document is based on the Student Accountability Information System (SAIS) Student Database Transaction Requirements document written by the Arizona Department of Education (<http://www.azed.gov/student-accountability/files/2011/12/sais-transaction-usage.pdf>).

Scope

While the Ed-Fi REST API handles a wide array of operations that are beyond the scope of a SIS, this document focuses on only those operations that are most common from a SIS.

SIS Operations

Assumptions

1. The schools and related calendars must be created and in place before the beginning of the school year.
2. The student record must already be present as evidenced by the fact that the unique student Id is required for the operations below. The students can be accessed through the */students* resource.
3. The term Student refers to an enrolled student (unless otherwise noted, i.e. student was withdrawn)

Note about read-only entities

For the initial release of the Ed-Fi REST API, the underlying data source does not contain base entities for education organizations such as LEAs and schools. It is up to the vendor to supply that information via REST calls. In a future release, these entities will be loaded by the state and will be read-only to vendors. There are other entities that follow this pattern including, but not limited to, types & descriptors and state course codes.

Operations

First time enrollment in a school.

Add a student school association.

Resources

POST /studentSchoolAssociations

Student returns to same school at beginning of new school year.

Add a student school association.

Resources

POST /studentSchoolAssociations

Student enrollment submitted in error.

Remove the incorrect student school association. If any related records were added such as student section associations, grades or attendance events, those will have to be removed before the student school association. Finally, add the correct student school association.

Resources

DELETE /studentSchoolAssociations/{id}
 POST /studentSchoolAssociations

Reported enrollment date is wrong.

Modify the existing student school association with the correct enrollment date.

Resources

POST /studentSchoolAssociations/{id}

Student registers for first time in this school in new school year and fails to attend.

Remove the incorrect student school association. If any related records were added such as student section associations, grades or attendance events, those will have to be removed before the student school association.

Resources

DELETE /studentSchoolAssociations/{id}

Student is readmitted to same school.

Add a new student school association.

Resources

POST /studentSchoolAssociations

Student readmission submitted in error.

Remove the incorrect student school association. If any related records were added such as student section associations, grades or attendance events, those will have to be removed before the student school association.

Resources

DELETE /studentSchoolAssociations/{id}

Student withdraws from school. Student NOT in a program or service.

Update the student school association with an ExitWithdrawDate and ExitWithdrawTypeDescriptorId.

Resources

PUT /studentSchoolAssociations/{id}

Student withdraws from school. Student is active in a program and will be exiting the program.

Update the student program association with an EndDate and then update the student school association with an ExitWithdrawDate and ExitWithdrawTypeDescriptorId.

Resources

PUT /studentProgramAssociations/{id}

PUT /studentSchoolAssociations/{id}

Reported withdrawal date is wrong for a student school association.

Update the student school association with the correct ExitWithdrawDate and ExitWithdrawTypeDescriptorId.

Resources

PUT /studentSchoolAssociations/{id}

Withdrawal submitted in error.

Update the student school association with null values for ExitWithdrawDate and ExitWithdrawTypeDescriptorId.

Resources

PUT /studentSchoolAssociations/{id}

Student absent during one day for a single reason.

Add a student school attendance event to log the absence.

Resources

POST /studentSchoolAttendanceEvents

Student absent different sections in a single day for different reasons.

Add a student section attendance event for each section missed.

Resources

POST /studentSectionAttendanceEvents

Student absent for consecutive school days for same reason.

Add a student school attendance event for each day missed.

Resources

POST /studentSchoolAttendanceEvents

Absence reason code for a submitted school absence date needs to be changed based on new information.

Update the student school attendance event with the correct information.

Resources

PUT /studentSchoolAttendanceEvents/{id}

Personal information is wrong or has changed.

Update the student with the correct information.

Resources

PUT /students/{id}

A student needs to be added but will not have an association.

Create a student with associated personal information without assigning the student to a school.

Resources

POST /students

The grade entry date is wrong.

Update the student school association with the correct EntryDate.

Resources

PUT /studentSchoolAssociations/{id}

The old grade level is incorrect and needs to be changed.

Update the student school association with the correct EntryGradeLevelTypeId.

Resources

PUT /studentSchoolAssociations/{id}

Correction made to attendance start date.

Update the student school attendance event with the correct EventDate.

<p><i>Resources</i></p> <p>PUT /studentSchoolAttendanceEvents/{id}</p>
<p>Student attendance submitted in error.</p> <p>Remove the incorrect student school attendance event.</p> <p><i>Resources</i></p> <p>DELETE /studentSchoolAttendanceEvents/{id}</p>
<p>Student whose membership lasted through scheduled last day of school reports that he/she will not attend this school this year and supplies a reason.</p> <p>Modify the student school association with an withdraw date and reason.</p> <p><i>Resources</i></p> <p>PUT /studentSchoolAssociations/{id}</p>
<p>Correct Student Membership without affecting personal information or district of residence</p> <p>Update student membership. (Equivalent to SAIS 006 Student Membership Change transaction).</p> <p><i>Resources</i></p> <p>PUT /studentSchoolAssociations/{id}</p>
<p>Change Student Funded District Of Residence, Student Full Time Equivalency, Student Grade Transfer and Student Payer Factors</p> <p>(Equivalent to SAIS 007 Student DOR Transfer, 008 Student FTE, 009 Student Grade Transfer, 010 Student Payer Factors transactions)</p> <p><i>Resources</i></p> <p>PUT /studentSchoolAssociations/{id}</p>
<p>Course created at a school</p> <p><i>Resources</i></p> <p>POST /courses</p> <p>POST /courseOfferings</p>
<p>Course changed at a school</p> <p><i>Resources</i></p> <p>PUT /courses {id}</p>

<p>PUT / courseOfferings {id}</p>
<p>Course deleted at a school</p> <p><i>Resources</i></p> <p>DELETE / sections {id}</p> <p>DELETE / courseOfferings{id}</p> <p>DELETE / courses {id}</p>
<p>Section assigned to a course or Section created at a school</p> <p><i>Resources</i></p> <p>POST / sections</p>
<p>Section deleted at a school</p> <p><i>Resources</i></p> <p>DELETE / sections {id}</p>
<p>Section changed at a school</p> <p><i>Resources</i></p> <p>PUT / sections{id}</p>
<p>Teacher assigned to section</p> <p><i>Resources</i></p> <p>POST / TeacherSectionAssociations</p>
<p>Teacher assigned to section changed</p> <p><i>Resources</i></p> <p>PUT / TeacherSectionAssociations {id} on the old teacher followed by</p> <p>POST / TeacherSectionAssociations on the new teacher</p>
<p>Student added to section</p> <p><i>Resources</i></p>

POST /StudentSectionAssociations
Student removed from section <i>Resources</i> PUT /StudentSectionAssociations {id}
Student receives grade mark <i>Resources</i> POST /grades
Student receives credits earned <i>Resources</i> PUT / studentAcademicRecord {id}